Pharmacy Portal

*Professor **Yanilda Peralta Ramos***
*Submission Deadline: **05/05/2025***
*Presentations: **05/12/2025** and **05/14/2025***

Project Description

Develop a web-based platform using PHP and MySQL for Pharmacists and Patients to be able to securely access prescription information tools and medication management. The pharmacy staff should also be able to manage inventory and sales effectively.

- You are provided with the starter code in PHP. Your task will be to connect this platform to a MySQL database and complete the required functionality so that the platform runs seamlessly.
- Create a **Report** explaining your code and the platform.
- Upload your source code (public) along with the report in PDF format to **GitHub** and submit the link to **Blackboard**.
- Make a Zoom live **presentation** to showcase your project on the presentation dates.

Guidelines

Starter Code: GitHub Link
Download and adapt the files for your implementation.

Requirements

Platform Features

1. The system must verify whether a pharmacist or patient already exists in the Users table before providing access. If the user does not exist, they should be added automatically.

2. New users can be added either directly or during the process of accessing prescription information.

3. Integrated user data will enable tracking and managing interactions, including prescription details, medication management, and inventory control.

MySQL Implementation (45 Points)

Database Setup

1. Create a database named pharmacy_portal_db. (5 Points)
2. Use the previously created database.

MySQL Tables and Relationships

1. Users Table (5 Points):
    - userId INT NOT NULL UNIQUE AUTO_INCREMENT
    - userName VARCHAR(45) NOT NULL UNIQUE
    - contactInfo VARCHAR(200) (stores email or phone number)
    - userType ENUM('pharmacist', 'patient') NOT NULL
    - Primary Key: userId
2. Medications Table (5 Points):
    - medicationId INT NOT NULL UNIQUE AUTO_INCREMENT

- medicationName VARCHAR(45) NOT NULL
- dosage VARCHAR(45) NOT NULL
- manufacturer VARCHAR(100)
- Primary Key: medicationId

3. Prescriptions Table (5 Points):
    - prescriptionId INT NOT NULL UNIQUE AUTO_INCREMENT
    - userId INT NOT NULL
    - medicationId INT NOT NULL
    - prescribedDate DATETIME NOT NULL
    - dosageInstructions VARCHAR(200)
    - quantity INT NOT NULL
    - refillCount INT DEFAULT 0
    - Primary Key: prescriptionId
    - Foreign Key: userId references Users(userId)
    - Foreign Key: medicationId references Medications(medicationId)
4. Inventory Table (5 Points):
    - inventoryId INT NOT NULL UNIQUE AUTO_INCREMENT
    - medicationId INT NOT NULL
    - quantityAvailable INT NOT NULL
    - lastUpdated DATETIME NOT NULL
    - Primary Key: inventoryId
    - Foreign Key: medicationId references Medications(medicationId)
5. Sales Table (5 Points):
    - saleId INT NOT NULL UNIQUE AUTO_INCREMENT
    - prescriptionId INT NOT NULL
    - saleDate DATETIME NOT NULL
    - quantitySold INT NOT NULL

- saleAmount DECIMAL(10, 2) NOT NULL
- Primary Key: saleId
- Foreign Key: prescriptionId references Prescriptions(prescriptionId)

Stored Procedures Views and Trigger:

1. AddOrUpdateUser(5 Points):
    - Purpose: Manages the addition of new users or updating existing users in the database.
    - Inputs: User ID (optional), username, contact information, user type.
    - Outputs: Updates or inserts user data accordingly.
2. ProcessSale(5 Points):
    - Purpose: Handles medication sales, updates inventory, and records sales transactions.
    - Inputs: Prescription ID, quantity sold.
    - Outputs: Updates inventory and logs the sale.

View

MedicationInventoryView (5 Points):

- Purpose: Provides an aggregated view of medication details alongside their current inventory levels.
- Contents: Medication name, dosage, manufacturer, and quantity available.

Trigger

AfterPrescriptionInsert (5 Points):

- Purpose: Updates inventory automatically following the insertion of a new prescription.
- Action: Reduces inventory based on the prescription quantity and notifies if stock is low.

Data Population

Populate each table with at least **three entries**. (5 Points)

PHP Implementation (45 Points)

- Update Database Connection Details:

  Update the database connection settings in the PharmacyDatabase.php file to match your MySQL server information.

- Test the Platform:

  Run the platform on your local PHP server. If the platform is functional and displays the homepage successfully. Test the addPrescription and ViewPrescription (2.5 Points)

- Create and Implement Medication Addition:

  Implement the method addMedication(...) in the PharmacyDatabase.php file to insert prescriptions into the database. (2.5 Points)

- Create and implement additional functions (5 Points)

- Complete and Implement User Addition:

  Complete and implement the method addUser($userName, $contactInfo, $userType) in the PharmacyDatabase.php file to add users to the database. (5 Points)

- Verify Medication Inventory View Functionality:

Verify that the MedicationInventoryView is functioning correctly, displaying accurate information about medication inventory. (5 Points)

- Complete and Implement User Data Retrieval:

  Complete and implement the method getUserDetails($userId) in the PharmacyDatabase.php file to retrieve user details, including prescriptions and user type. (5 Points)

- Create Secure Login System:

  Implement a secure login system in the login.php file for patients and pharmacists with separate access levels. This should include:

  Secure password hashing and verification.

  Session management to handle login states securely.

  Role-based access control to differentiate functionality between pharmacists and patients. (5 Points)

Publishing (5 Points)

1. Write a **Report** explaining your code and its functionality. (10 Points)

Upload your source code, report (PDF), and SQL scripts to a public GitHub repository. Include MySQL queries. (5 Points)

2. Submit the repository link on Blackboard and include screenshots in the report appendix.

Presentation (5 Points) – Presentation mandatory to receive a grade for this project

Present your completed project live on Zoom on **05/12/2025** or **05/14/2025**.